GausHitFinder Update

(...that's Gauss with one s)

Jonathan Asaadi

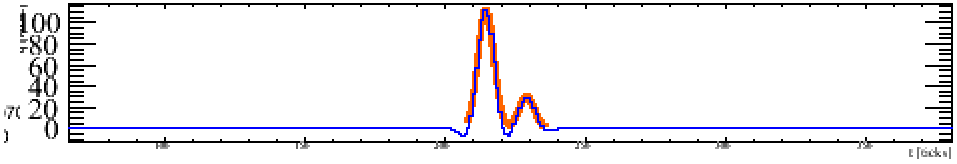
Syracuse University

09/05/2012

Gauss Rifle produces one kind of hit... :-(

Gauss Trojan Horse virus will hit your bank account... :-(

Gaus hit's will help you do your physics analysis :-)



Palm Tree Approved

Reminder of what we've shown before...

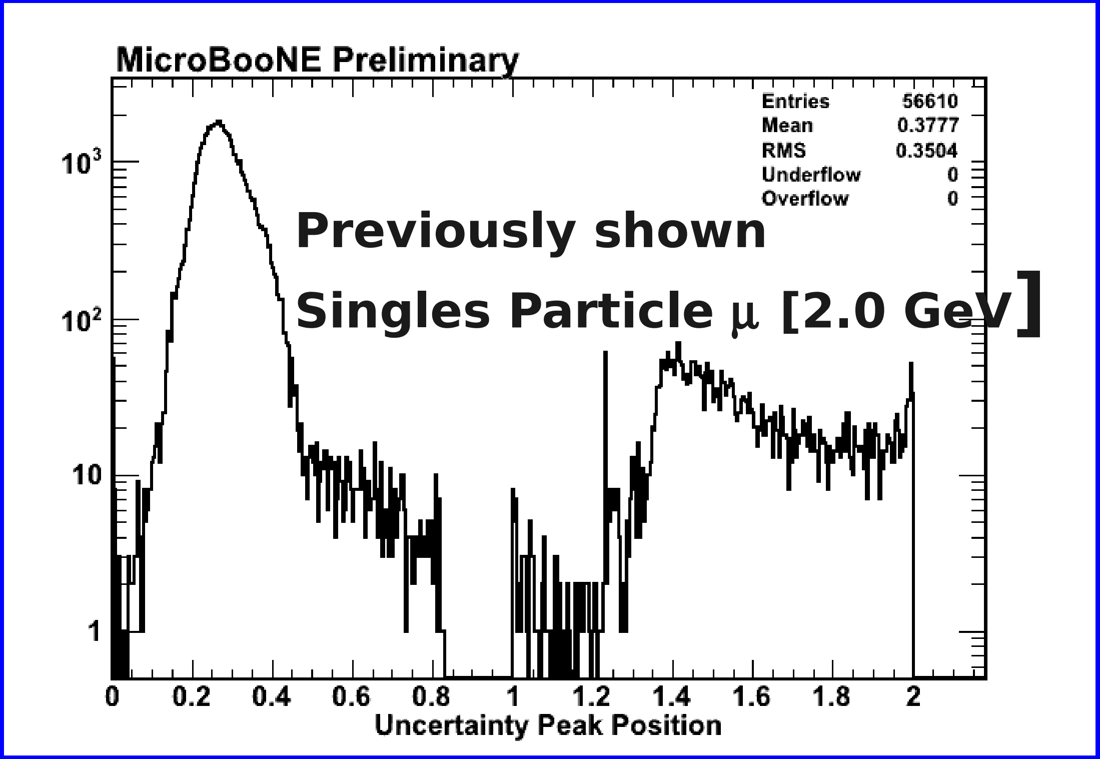
- GausHitFinder reconstructs >99% of the same hits as FFTHitFinder
- GausHitFinder agrees with FFTHitFinder on various hit parameters (Peak Position, Hit Width, “Charge”, etc...)
- GausHitFinder has very similar time performance with FFTHitFinder
- GausHitFinder correctly reports back the hit multiplicity (number of peaks fit to a hit, which was previously hard coded to 1 in FFTHitFinder)
- Comparison study with HitCheater package showed > 93% efficiency for GausHitFinder hits
- Added a parameter to the .fcl file that allows user to cut on the χ^2 of the fitted hit

See Back-up slides for more details

Remaining Questions

- How are the errors calculated for the hits (and hence the χ^2 used in the cut)
- What do the “Pull” values of the peak position error look like

Pull = (Peak Position$_{\text{Truth}}$ – Peak Position$_{\text{Reco}}$) / $\sigma_{\text{Peak Postion}}$



Caution:
Results coming were a little rushed at the very end...so I am showing results for hits from single muon gun with multiplicity = 1...I will update this talk later with a full set of results for both Genie events as well as hit with multiplicity > 1

GausHitFinder Code

How the errors are calculated

- Previously, the error associated with the fit was just the chi2 minimization over the range of the fit and the error on the bins was just sqrt(bin content)

$$\frac{f(x_i,\alpha)-e_i)^2}{\sigma_i^2}$$

See the following link for my best understanding...

http://wwwasdoc.web.cern.ch/wwwasdoc/minuit/node7.html

Beware:

Code Snippets below

- **hitSignal.Fit(&Gaus,”QNRW”,””,startTime, endT);**

  Q = Quiet mode for fit, N = Don't draw, R = Use range specified, W = Set weight for non-empty bins = 1

  PeakError = Gaus.GetParError(1);

- However, I realized that there are better hooks in ROOT for the errors I'm computing here...

GausHitFinder Code

How the errors are calculated

- I've updated GausHitFinder to do a better job handling the errors by first using **hitSignal.Sumw2()** to force the computation of the sum of the square of the weights per bin and thus the error on a bin is the sqrt(sum of squares of weights)
- Secondly, I've changed how the errors of the fit are calculated.
  - First finding the normalization, mean, and RMS of a Gaussian for the hit just as before
  - Now using this Gaussian as a seed for a second Gaussian fit to find the errors and chi2 using:

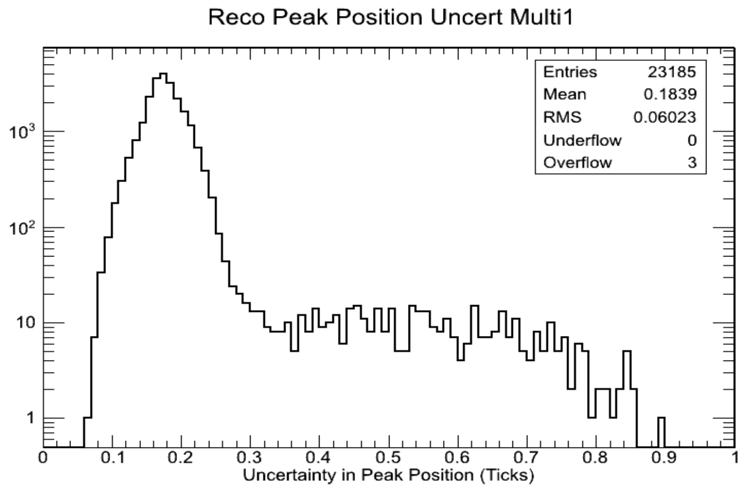
    hitSignal.Fit(hit,”QNRLLi”,””,StartTime,EndTime);

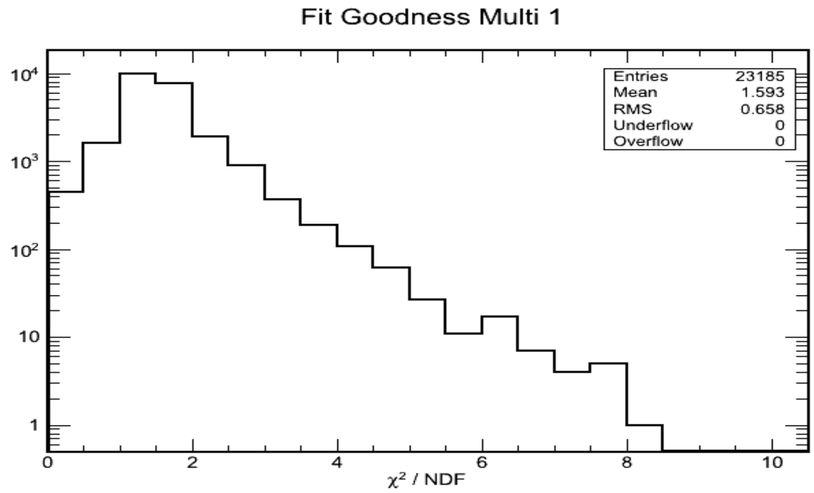
    Q = Quiet mode for fit, N = Don't draw, R = Use range specified,

    LL = Improved Log-likelihood fit, I = Use the integral of function in bin instead of the value at the bin center

GausHitFinder

Peak Position Error





$$\text{Pull} = (\underbrace{\text{Peak Position}_{\text{Truth}}} – \text{Peak Position}_{\text{Reco}}) / \sigma_{\text{Peak Postion}}$$

Truth peak position is obtained by using art::ServiceHandle<cheat::BackTracker>

Many thanks to B. Rebel for helping me figure this out...

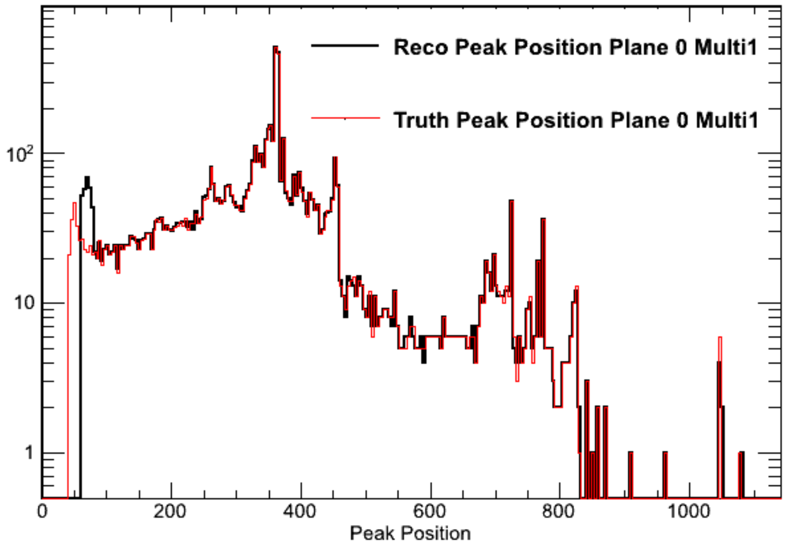
1) Take the Reco'd hit and use the HitToXYZ(hit) function to obtain the truth xyz position

2) Match that truth xyz position to the nearest channel using the NearestChannel(xyz,plane) function

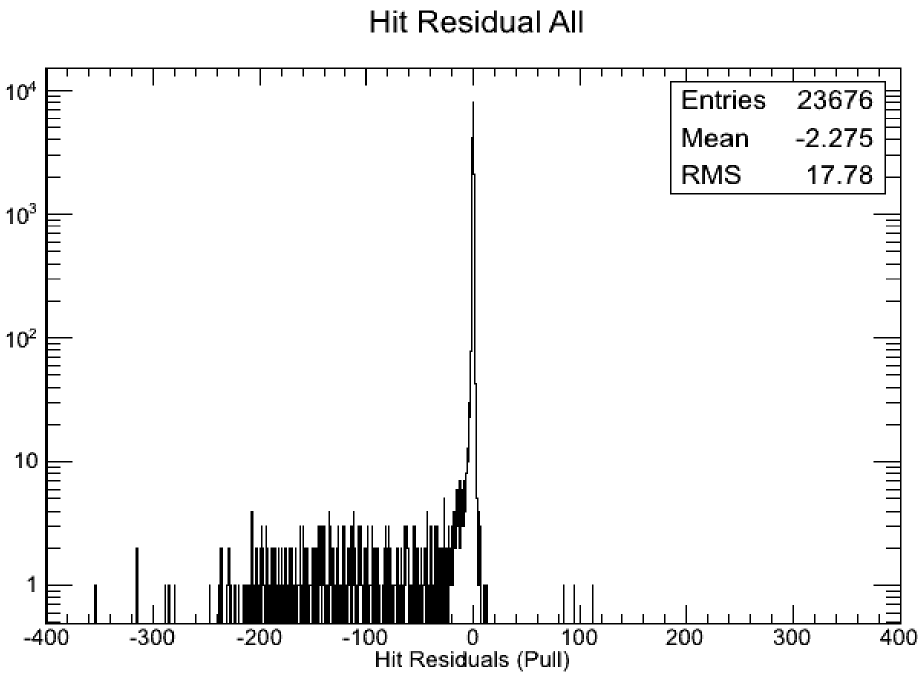
3) Finally use the ConvertXToTicks(xyz[0],plane,tpc,channel) to get back what the truth peak position in ticks

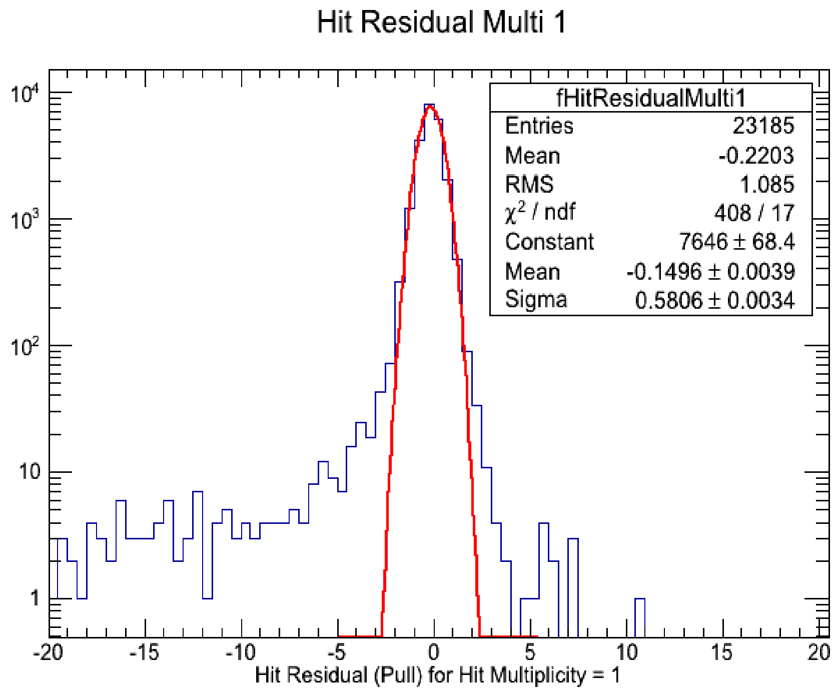
GausHitFinder

Pull Values



Case example showing that the peak position for the truth and reco are very close to one another for the hits (similar results for all the other planes)

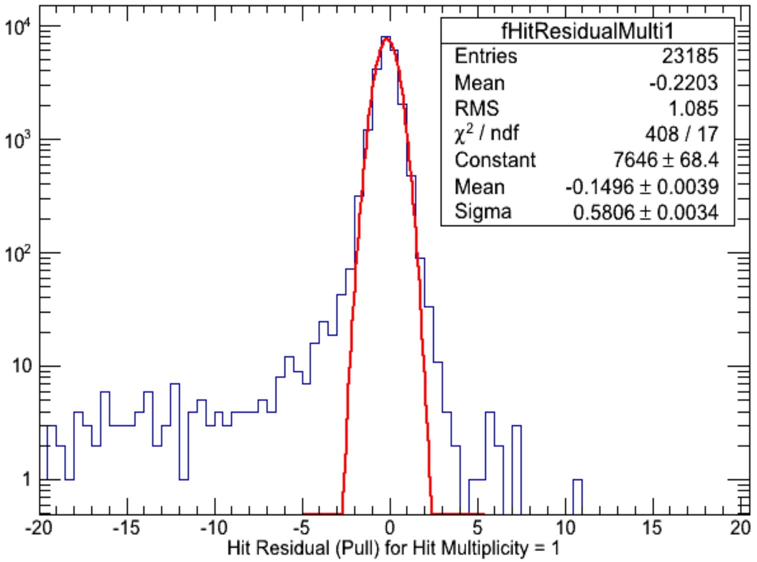




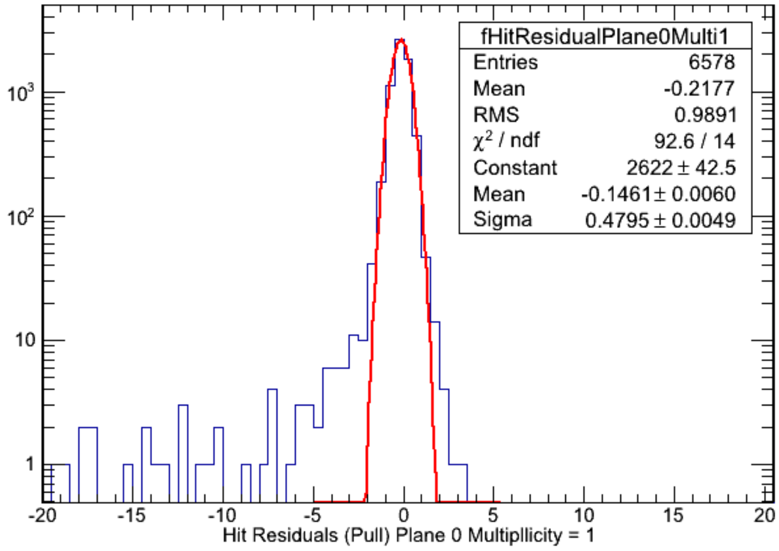
GausHitFinder

Pull Values

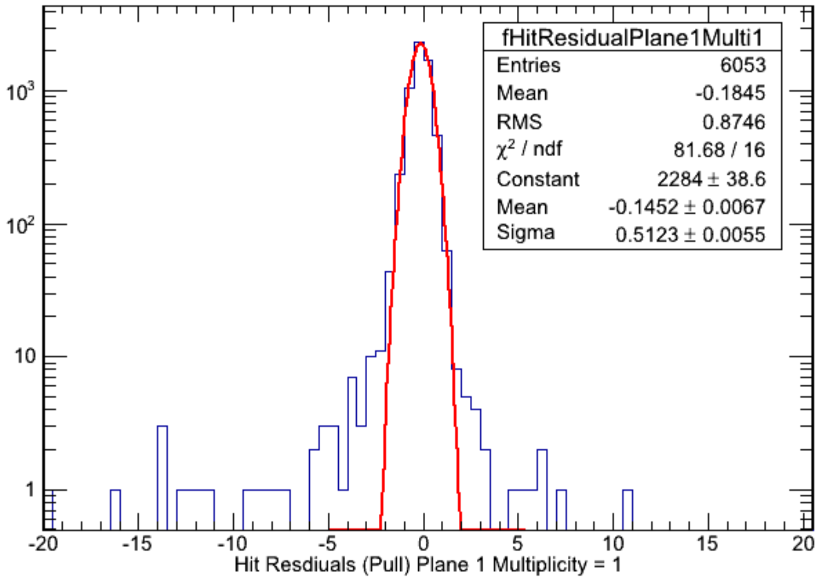
Hit Residual Multi 1



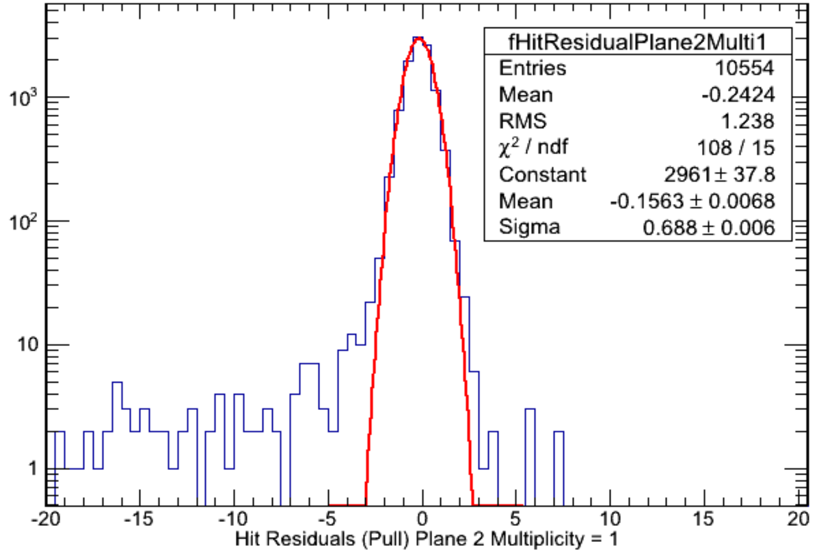
Hit Residual Plane 0 Multi 1



Hit Residual Plane 1 Multi 1



Hit Residual Plane 2 Multi 1



Next Steps

- Finish and update this talk to include all hits and has Genie events present
- Follow-up on the results from the Pull values presented here
- Clean-up and check in updated versions of GausHitFinder and GausHitFinderAna which has the fixes and histograms presented here for general use

Back-up Slides

Results of the GausHitFinder Algorithm

Single Particle μ [2.0 GeV] and Genie Events

What is the time performance of the two algorithms?

FFTHitFinder

(Running over 10 single muon events)

Avg. Time
~ 14 seconds per event

GausHitFinder

(Running over 10 single muon events)

Avg. Time
~ 14 seconds per event

FFTHitFinder

(Running over 10 Genie events)

Avg. Time
~ 13 seconds per event

GausHitFinder

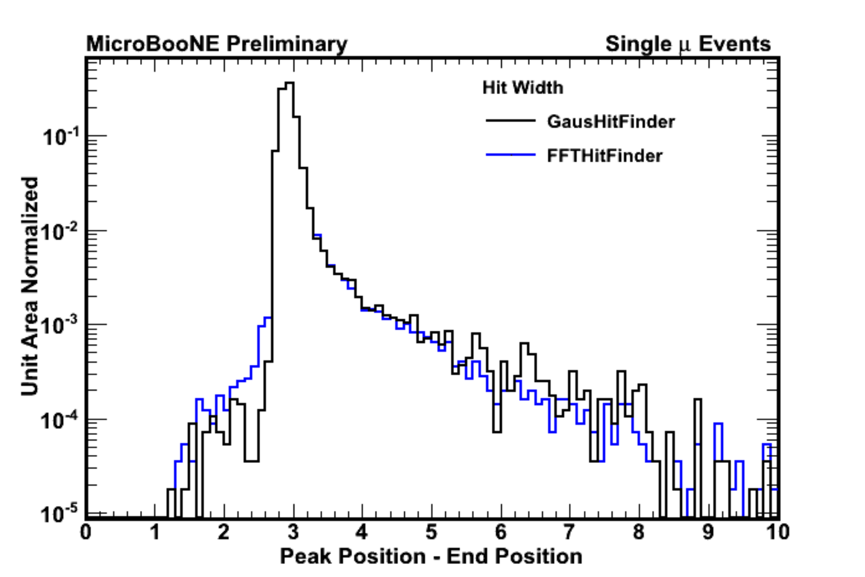
(Running over 10 Genie events)

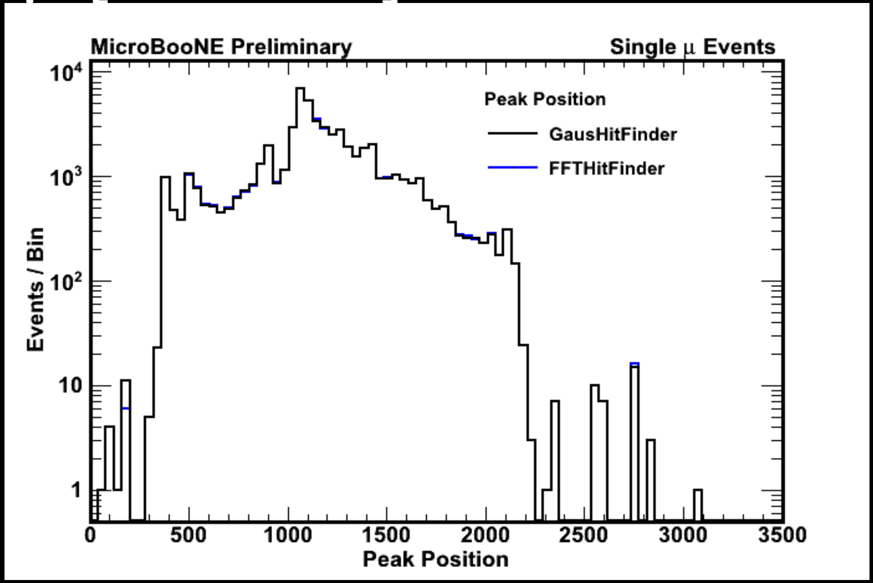
Avg. Time
~ 13 seconds per event

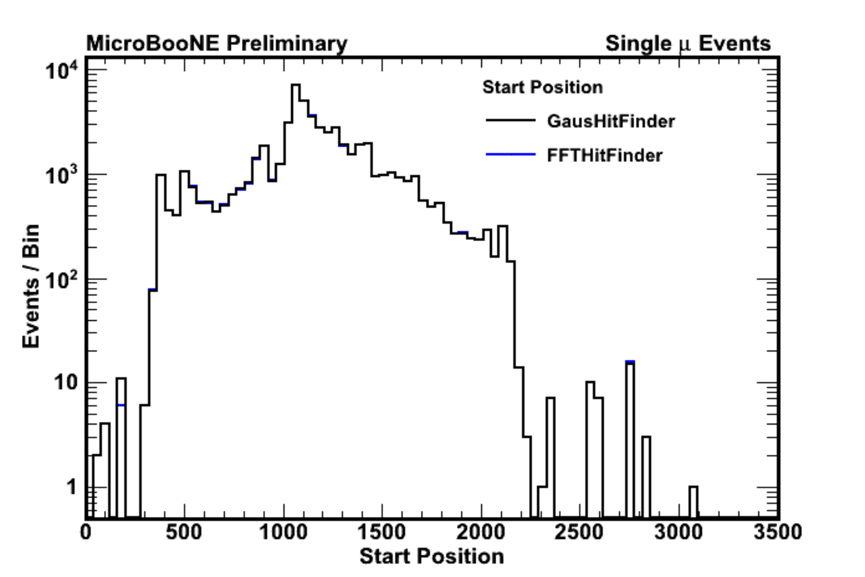
No Time Performance Difference!

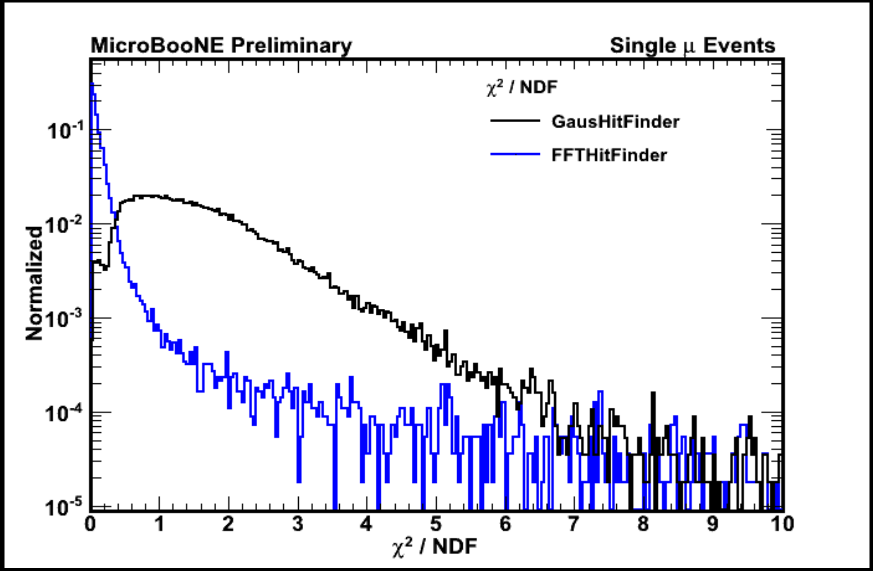
Comparing GausHitFinder vs FFTHitFinder

Single Particle μ [2.0 GeV]





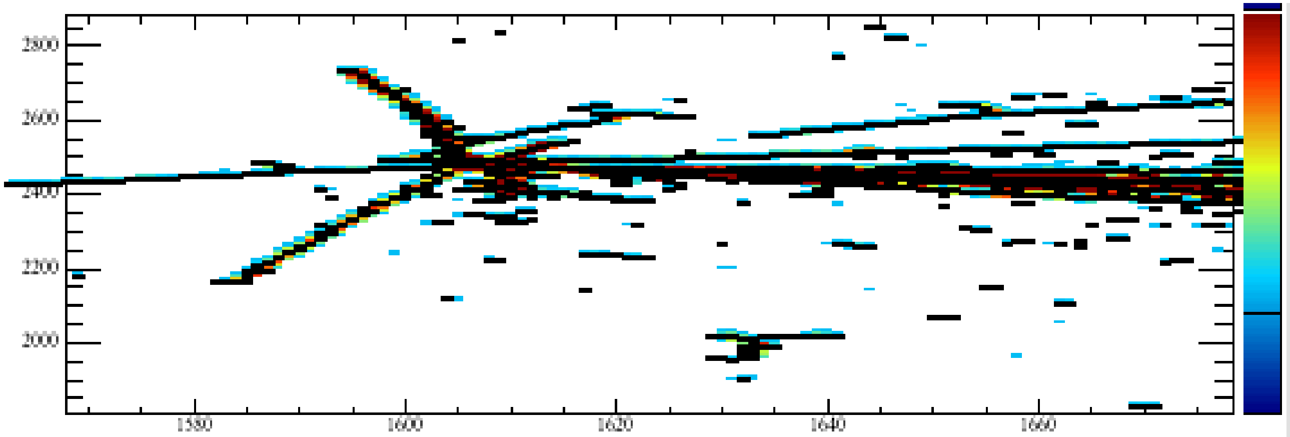


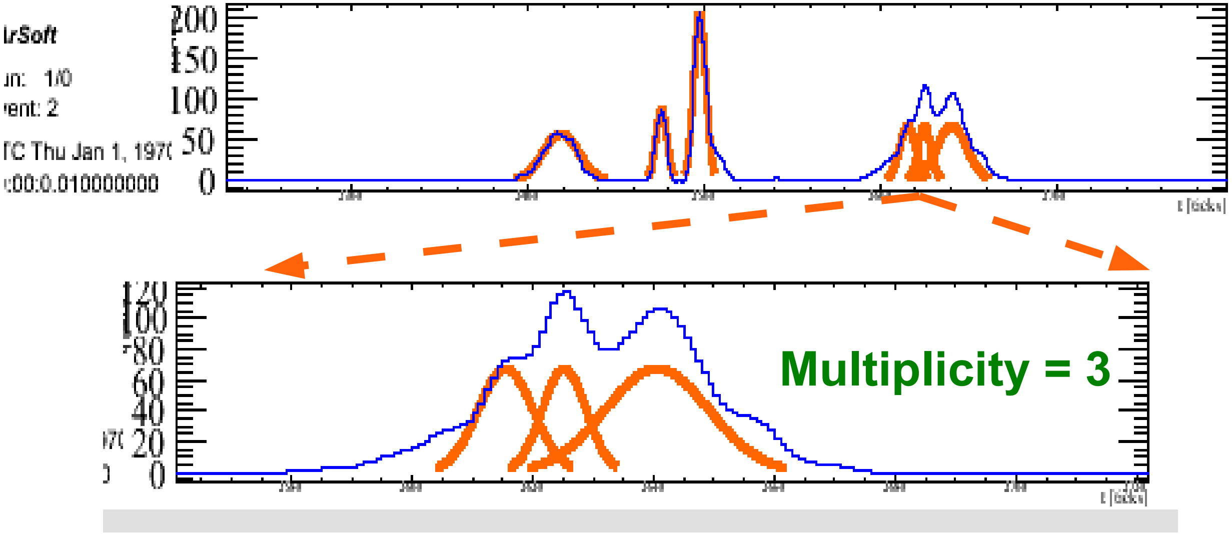


Results of the GausHitFinder Algorithm

Genie Events

October - December 2012
multiplicity
the must-have magazine for all moms of multiples.
What to do when they ask why?
Yes, you can!
Capture moments without a meltdown
+
Hottest holiday toys for multiples
Survive & keep the miracle alive!
www.multiplicitymag.com





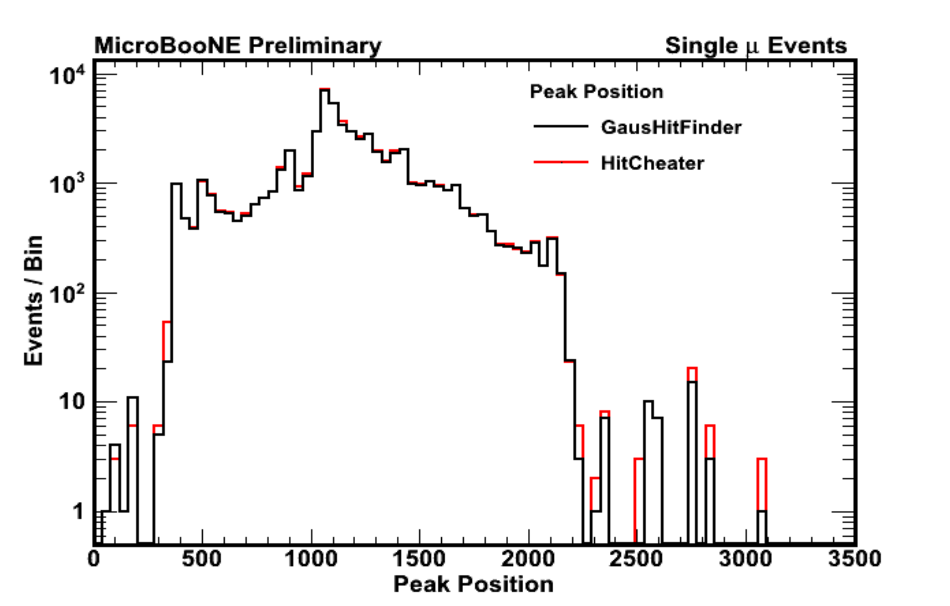
Multiplicity = 3

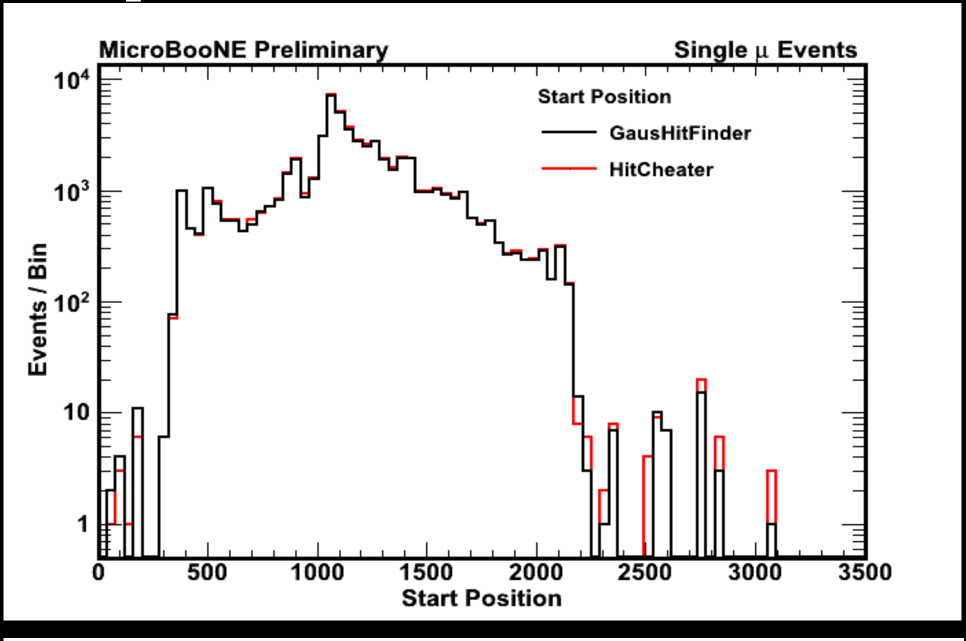
GausHitFinder now can handle multi-peaked hits as well as correctly identify the multiplicity of the hit

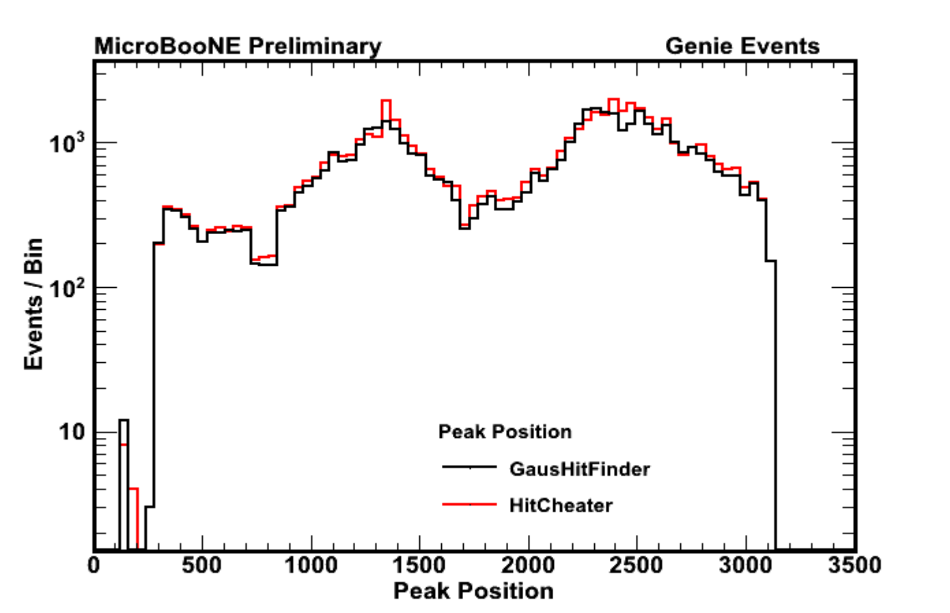
→ Can allow us to identify how to handle “Goodness of fit” for high multiplicity hits

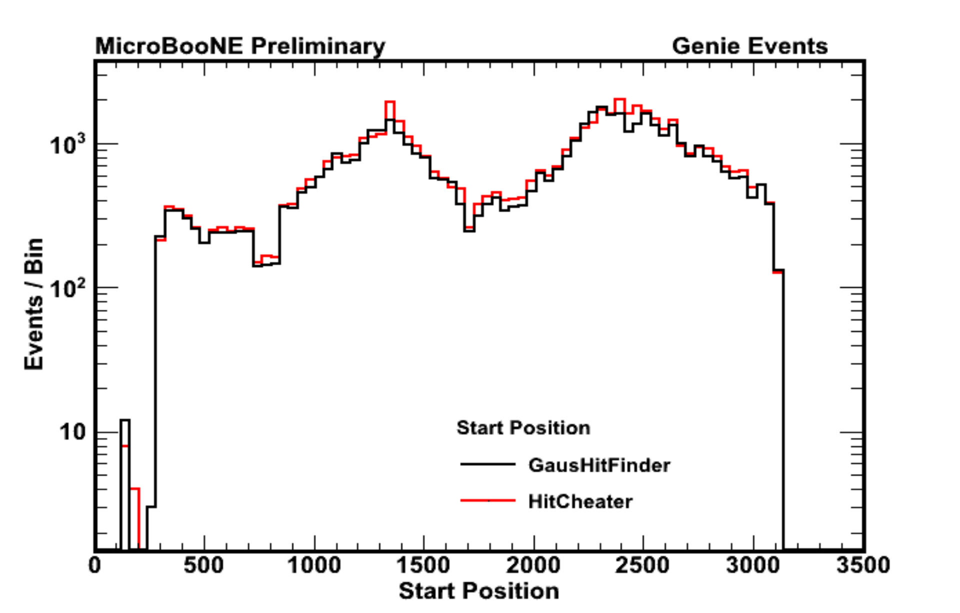
Comparing GausHitFinder vs HitCheater

Single Particle μ [2.0 GeV] and Genie Events









Comparing GausHitFinder vs HitCheater

Single Particle μ [2.0 GeV] and Genie Events

Single Particle μ [2.0 GeV]

HitCheater	GausHitFinder
(Running over 10 single muon events)	*(Running over 10 single muon events)*
Number of Hits Found = 58,047	**Number of Hits Found = 56,891**

“Efficiency” = 98%

Genie Events

HitCheater	GausHitFinder
(Running over 10 Genie events)	*(Running over 10 Genie events)*
Number of Hits Found = 53,195	**Number of Hits Found = 49,339**

“Efficiency” = 93%